United States Department of Agriculture

USDA

NRCS Natural Resources
Conservation Service

Soil Data Access Query Guide

Soil Survey Division
September 2017

Introduction

In order to effectively query any database, one needs to know the following:

What entities and attributes are represented in the database?

How is each of the entities represented in the database unambiguously identified?

How are the entities represented in the database related?

What data in the database can I depend on always being populated?

What is the syntax of the query that will extract the data I need?

This information is distributed between this document and other documents referenced within this document.

The Soil Data Mart Data Access suite of applications and web services allows a user to execute a query against the Soil Data Mart database in either real-time via a web service, or via a web application interface as either a real-time request whose results are returned to your web browser or as a queued request whose results are not returned in real-time. Further details about executing a query against the Soil Data Mart database are covered in the section titled "Executing Queries against the Soil Data Mart Database".

The Soil Data Mart Database

The Soil Related Entities in the Soil Data Mart Database

The aggregated data that characterizes a survey area was partially derived from point samples taken over the course of that survey. A morphological description was recorded for a number of sample points, also known as pedons. Soil samples taken at numerous points were sent to a laboratory for a variety of physical, chemical and sometimes mineralogical analyses.

This data is eventually aggregated into a number of map unit components that characterize the soils in that survey area. Therefore the entire set of data for a survey area includes both point and aggregated data.

At this time the Soil Data Mart database includes only the aggregated data for a survey area. The point data that was collected in order to produce that aggregated data is currently stored in what we refer to as the National Soil Survey Center (NSSC) Soil Survey Laboratory Soil Characterization Database (https://ncsslabdatamart.sc.egov.usda.gov/).

At some point in time we hope to provide an integrated database that includes both point and aggregated data for a survey area, but for the time being, no point data is available in the Soil Data Mart database.

The following four entities, survey area, map unit, map unit component and soil horizon or layer, make up what we refer to as the backbone of the aggregated soil data model. A survey area is composed of multiple map units. A map unit is composed of one or more map unit components. A map unit component is composed of one or more soil horizons or layers. Ultimately, virtually every attribute in the Soil Data Mart database is an attribute of one these four soil entities.

Survey Area

Definition

A soil survey area is a geographic region for which a soil map with a comprehensive legend exists. A soil survey area is also the administrative region for which a particular soil survey is conducted. A published soil survey area may be composed of all or part of several soil survey areas for which individual soil surveys were conducted.

It is not uncommon for a county and a soil survey area to share the same boundary. It is not uncommon for a soil survey area to encompass all or part of multiple counties. Soil survey is conducted in all 50 states, and for Caribbean and Pacific Basin US territories.

A goal for soil survey is to be relatively seamless (seamless map unit joins) across political boundaries, yet recognize the need for some natural boundaries to limit the extent of a particular soil. A Major Land Resource Area (MLRA) is the region by which soils are considered to be naturally bounded. In other words, while there is a desire to have seamless map unit joins across political boundaries, there is no imperative to have seamless soil map unit joins across MLRA boundaries.

Survey areas are mapped at a level of detail referred to as "order". Order ranges from 4 to 1, increasing in detail. Part of a survey area may be mapped at one level of detail and part of the survey area may be mapped at another level of detail. The typical NRCS SSURGO soil survey area is mapped at order 2, or approximately 1:24000. We also provide a soil map and corresponding data at a scale of approximately 1:250,000. We refer to this product as STATSGO. Both SSURGO and STATSGO data is stored in the

Soil Data Mart database. For details about how to distinguish between these two categories of aggregated soil data, please see the section titled "Discriminating between SSURGO and STATSGO".

What is the Significance of a Survey Area?

A survey area is an administrative region by which soil survey is conducted and by which aggregated soil survey data is distributed. The survey area by which a set of aggregated soil data is distributed may or may not be the same survey area, in whole or part, for which that data was originally collected.

With two possible exceptions, you probably have little or no need to deal with survey areas. What you really need to know is which map units coincide with your area of interest. Your area of interest may reside within a single survey area or may span all or part of multiple survey areas.

The at least three reasons that a survey area may be of concern are:

1. Our data is formally versioned by survey area. If for any reason you need to record the version of data that was used for a particular purpose, you would need to somehow permanently record the version of that survey area. For additional details on this subject, please see the section titled "Data Version and Citing the Soil Data Mart"

2. You may wish to constrain a query to an area of interest. Referencing survey areas that coincide with your area of interest is a relatively inaccurate way to constrain a query to an area of interest. For addition details on this subject, please see the section titled "Constraining a Query to an Area of Interest".

3. Fuzzy logic soil interpretations are included on a survey area basis. In other words, the exact same set of fuzzy logic soil interpretations will be available for every map unit component in a given survey area. What fuzzy logic soil interpretations are available for the map unit components in a given survey area is recorded in the table named "sainterp". For additional information about fuzzy logic soil interpretations, please see the section titled "Fuzzy Logic Soil Interpretations".

How is an Instance of a Survey Area Identified in the Database?

A record in the table named "legend" and records that repeat for a given survey area in other tables characterize a survey area. A survey can be reliably identified by its corresponding system generated ID, legend.lkey. Although this ID contains only digits,

the data type of legend.lkey is "variable character". No two map units in the Soil Data Mart database share the same value for mapunit.mukey.

A legend can also reliably be identified by the combination of its corresponding area type name, legend.areatypename, and its corresponding area symbol, legend.areasymbol. There are currently only two types of survey areas in the Soil Data Mart database, a SSURGO survey area and a STATSGO survey area. The area type name of a SSURGO survey area is always "Non-MLRA Soil Survey Area", and the area type name of a STATSGO survey area is always "Country". In the future it is possible that there may be other types of survey areas in the Soil Data Mart database. For additional information about this difference between SSURGO and STATSGO survey areas, please see the section titled "Discriminating between SSURGO and STATSGO".

In the Soil Data Mart database, the terms "survey area" and "legend" are synonymous. In some of our other information systems there can be more than one legend for a given survey area.

In the Soil Data Mart database, a record in the table named "sacatalog" also corresponds to a survey area. The reason why a survey area is represented by a record in two different tables is beyond the scope of this document. A record in the table named "sacatalog" can be reliably identified by its corresponding value of area symbol, sacatalog.areasymbol.

An astute reader might ask why the combination of the values in legend.areatypename and legend.areasymbol is required to identify a record in the table named "legend", but only the value in sacatalog.areasymbol is required to identify a record in the table named "sacatalog". This is because, at the current time, we just happen to know that no two survey areas in the Soil Data Mart database, SSURGO versus STATSGO notwithstanding, share the same value for area symbol. This could change in the future, which means that at that point we will have to add the area type name column to table "sacatalog".

What Attributes Can Repeat for a Survey Area?

Attributes that can repeat for a survey area include:

> Attributes that characterize a geographic coincidence between that survey area and some other type of geographic region, e.g. State or Territory, County or Parish, Major Land Resource Area, etc.

Map Unit

Definition

A detailed map for a soil survey area is composed of the following spatial components:

Soil Survey Area Boundary Polygon(s)
Map Unit Polygons
Line Map Units
Point Map Units
Line Spot Features
Point Spot Features

A map unit is represented by one or more polygons, lines or points where the management of the corresponding soils contained with that map unit is relatively consistent for the set of land uses for which the survey is intended.

Here is some of the discussion of the map unit concept from the Soil Survey Manual.

A map unit is a collection of areas defined and named the same in terms of their soil components or miscellaneous areas or both. Each map unit differs in some respect from all others in a survey area and is uniquely identified on a soil map. Each individual area on the map is a delineation.

…The kinds of map units used in a survey depend primarily on the purposes of the survey and the pattern of the soils and miscellaneous areas in the landscape. The pattern in nature is fixed and it is not exactly the same in each delineation of a given map unit. In soil surveys these patterns must be recognized and map units designed to meet the major objectives of the survey. It must be remembered that soil interpretations are made for areas of the land and the most useful map units are those that group similarities.

While studying the soil patterns in different landscapes, the soil scientist must keep in mind how best to relate the patterns to appropriate map units. The kinds of map units, the level of soil taxa, and the phases needed to satisfy the survey objectives must be determined. This requires many judgments. Every map unit that is tentatively identified is evaluated by two tests: (1) Can it be mapped consistently? (2) Is it needed to meet the objectives of the survey?

Designing map units to indicate significant differences in behavior among soils is particularly important to meet the current objectives of a survey. Reflecting differences in genesis and morphology is also important, even if no immediate differences in interpretations are known. Differences in soil properties that do not affect current interpretations may be important in the future; however, having too many delineations seriously reduces the immediate usefulness of a soil map. A potential benefit must be weighed carefully against the costs incurred in making additional separations. One

objective of every soil survey is to record knowledge about soils, but this does not mean that the soil map must show the location of every kind of soil in a survey area or that the publication must record all that has been learned about the soils.

What is the Significance of a Map Unit?

A map unit is the smallest soil entity that is ever delineated on a soil map. Map unit components are never delineated on a soil map. The position of a map unit component within its corresponding map unit can only be inferred from the geomorphology description associated with that map unit component.

Therefore determining the soils that occur in an area of interest can only be done indirectly by determining the map units that coincide with that area of interest. It then follows that the most accurate way to constrain a query to an area of interest is to reference the map units that coincide with that area of interest. For addition details on this subject, please see the section titled "Constraining a Query to an Area of Interest".

How is an Instance of a Map Unit Identified in the Database?

A record in the table named "mapunit" and records that repeat for a given map unit in other tables characterize a map unit. A map unit can be reliably identified by its corresponding system generated ID, mapunit.mukey. Although this ID contains only digits, the data type of mapunit,mukey is "variable character". No two map units in the Soil Data Mart database share the same value for mapunit.mukey.

The map units for a given survey area can be reliably distinguished by their corresponding map unit symbol, mapunit.mukey. It then follows that the combination of survey area type name, legend.areatypename, survey area symbol, legend.areasymbol and map unit symbol, mapunit.musym, can also be reliably used to identify a map unit. In other words, no two map units in the Soil Data Mart database share the same combination of values for legend.areatypename, legend.areasymbol and mapunit.musym.

When displaying the map units for a given survey area in a report, those map units are typically sorted in ascending order by map unit symbol, mapunit.musym, where a special type of sort is used to deal with the fact that map unit symbols, which are recorded as a variable character field, are often a combination of a number following by a string, or a string followed by a number, or a number followed by a string followed by a number. We refer to this type of sort as "numeric-lexigraphic-numeric". Any preceding or trailing numbers in the map unit symbol are sorted as numbers, and any character string portion is sorted lexigraphically. Note that map unit symbols are the only case sensitive attribute in the entire Soil Data Mart database, which is why we use a lexigraphic sort.

Since no sort equivalent to "numeric-lexigraphic-numeric" is available in SQL, we have saved the sequence number that resulted from such a sort in mapunit.museq, but keep in mind that this value can only be used to logically sequence the map unit symbols within a given survey area. In our applications, when we need to display a set of map units that span more than one survey area, we tend to display the map units for survey area one, followed by the map units for survey area two, and so on, rather than displaying all map units in a single list with no control breaks.

What Attributes Can Repeat for a Map Unit?

Attributes that can repeat for a map unit include:

Attributes related to the expected yields of crops that are typically grown on that map unit, for a high level of management.

Attributes that characterize a geographic coincidence between that map unit and some other type of geographic region, e.g. State or Territory, County or Parish, Major Land Resource Area, etc.

Various kinds of narrative text notes about that map unit.

Map Unit Component/Inclusion

Definition

Map units are composed of non-delineated soil taxa and miscellaneous areas. Soil taxa are soils that can be classified at some level of soil taxonomy. In a typical order 2 survey, the soils that make up a map unit are typically classified at the series or phase level, but keep in mind that the range of characteristics of a component is permitted to extend beyond the range of the corresponding named series. In higher order surveys, the soil taxa may be classified at the family level or higher.

The soil taxa in a map unit are not delineated on the map. Relative position of different soil taxa may be connoted by the geomorphic description of where each soil occurs on the landscape.

A map unit typically contains major and minor components, and the percent composition of each component is noted. A minor component may be referred to as an "inclusion". Minor components may not be described at the same level of detail as major components. It is not uncommon for a minor component to not represent a named soil series.

The mix of components in a map unit determines map unit kind. From the Soil Survey Manual:

Consociations - In a consociation, delineated areas are dominated by a single soil taxon (or miscellaneous area) and similar soils. As a rule, at least one half of the pedons in each delineation of a soil consociation are of the same components that provide the name for the map unit. Most of the remainder of the delineation consists of soil components so similar to the named soil that major interpretations are not affected significantly.

Complexes and associations - Complexes and associations consist of two or more dissimilar components occurring in a regularly repeating pattern. Only the following arbitrary rule related to mapping scale determines whether the name complex or association should be used. The major components of a complex cannot be mapped separately at a scale of about 1:24000. The major components of an association can be

separated at a scale of about 1:24000. In either case, the major components are sufficiently different in morphology or behavior that the map unit cannot be called a consociation. In each delineation of either a complex or an association, each major component is normally present, though their proportions may vary appreciably from one delineation to another.

Undifferentiated groups - Undifferentiated groups consist of two or more taxa components that are not consistently associated geographically and, therefore, do not always occur together in the same map delineation. These taxa are included as the same named map unit because use and management are the same or very similar for common uses.

What is the Significance of a Map Unit Component?

When we refer to the soils that occur in an area of interest, we are talking about map unit components, although a map unit component may also be a non-soil area, i.e. a miscellaneous area. A map unit component is the entity for which the vast majority of empirical and interpretative soil attributes are recorded. Although not typical, some of the attributes recorded for a map unit component that corresponds to a soil may also be recorded for a non-soil map unit component.

How is an Instance of a Map Unit Component Identified in the Database?

A record in the table named "component" and records that repeat for a given map unit component in other tables characterize a map unit component. A map unit component can only be reliably identified by its corresponding system generated ID, component.cokey. This ID is a combination of digits and characters. No two map unit components in the Soil Data Mart database share the same value for component.cokey.

Much of the time the components for a given map unit can be distinguish by the combination of component name, component.compname, local phase name, component.localphase, representative percent composition, component.comppct_r, and representative slope, component.slope_r, but this is not 100% reliable, especially since surface texture is sometimes also used to distinguish one map unit component from another. Surface texture is not an attribute found in the tabled named "component". For additional details on the concept of representative values, please see the section titled "High, Low and Representative Values".

When displaying the components for a given map unit in a report, those components are typically sorted by representative percent composition, component.compct_r, descending,

component name, component.compname, ascending, and local phase name, component.localphase, ascending.

What Attributes Can Repeat for a Map Unit Component?

Attributes that can repeat for a map unit component include:

Attributes related to the expected yields of crops that are typically grown on that map unit component, for a high level of management.

Features which are diagnostic in determining the soil taxonomic classification of that map unit component.

Ecological sites and other vegetative classifications that are associated with that map unit component.

The types and degree of erosion typically associated with that map unit component.

The kinds of plants that are typically found on that map unit component.

The expected productivity of trees that are typically harvested from that map unit component.

Geomorphic features and attributes that characterize the position of that map unit component in the landscape.

Land use specific interpretations for that map unit component.

Various temporal attributes that characterize that map unit component, e.g. climate data, soil moisture and soil temperature. Much of this data is recorded by month.

The parent materials from which that map unit component formed.

Trees suitable for a windbreak on that map unit component.

Root limiting soil restrictions that are typically associated with that map unit component.

Surface fragments that are typically found on that map unit component.

Trees that are recommended for growing on that map unit component.

Soil Horizon/Layer

Definition

Here is the definition of a soil horizon and layer from the Soil Survey Manual:

A soil horizon is a layer, approximately parallel to the surface of the soil, distinguishable from adjacent layers by a distinctive set of properties produced by the soil-forming process. The term layer, rather horizon, is used if all of the properties are believed to be inherited from the parent material or no judgment is made as to whether the layer is genetic.

A map unit component is typically composed of more than one soil horizon or layer. There is no theoretical limit on the number of soil horizons or layers that may exist for a map unit component, but any increasing number above six becomes increasingly unlikely.

What is the Significance of a Soil Horizon or Layer?

The majority of empirical morphological, chemical and physical soil properties of a map unit component are associated with a soil horizon or layer.

How is an Instance of a Soil Horizon or Layer Identified in the Database?

A record in the table named "chorizon" and records that repeat for a given soil horizon or layer in other tables characterize a soil horizon or layer. A soil horizon or layer can only be reliably identified by its corresponding system generated ID, chorizon.chkey. This ID is a combination of digits and characters. No two soil horizons or layers in the Soil Data Mart database share the same value for chorizon.chkey.

Much of the time the soil horizons or layers for a given map unit component can be distinguish by their representative depth to top, chorizon.hzdept_r, and representative depth to bottom, chorizon.hzdepb_r, but this is not 100% reliable. Transitional soil horizons, those with a slash character in their horizon nomenclature, are actually entered as two separate records with the exact same depth values. For additional details on the concept of representative values, please see the section titled "High, Low and Representative Values".

When displaying the soil horizons or layers for a given map unit component in a report, those soil horizons or layers are typically sorted in ascending order on representative depth to top, chorizon.hzdept_r.

What Attributes Can Repeat for a Soil Horizon or Layer?

Attributes that can repeat for a soil horizon or layer include:

AASHTO soil classification.

Attributes related to soil consistence.

Attributes related to embedded fragments.

Attributes related to pores.

Attributes related to soil structure.

Attributes related to soil texture or terms used in lieu of texture.

Unified soil classification.

How Can I Distinguish Between a Soil Horizon and a Soil Layer?

Our original source data did not make this distinction. At some point in the evolution of aggregated soil database, attributes to record soil horizon nomenclature were added, but those attributes tend to only be populated for more recent updates of a survey area.

For a record in table chorizon for which the horizon nomenclature attributes are not populated, one cannot assume that record corresponds to a soil horizon. When the horizon nomenclatures attributes are populated, the corresponding record may or may not correspond to a soil horizon.

If the attribute chorizon.hzname contains a designation for a single soil horizon, you can assume that record does correspond to a soil horizon. If the attribute chorizon.hzname contains more than one soil horizon designation, that record corresponds to a layer containing multiple soil horizons. Our original intent was that chorizon.hzname should never contain more than one soil horizon designation, but in some cases our soil data stewards were more creative than we anticipated. This is an issue that will eventually need to be cleaned up.

What Tables can I Query?

Introduction

A SQL statement may select from only a subset of the existing tables in the Soil Data Mart database. In addition, a SQL statement may also reference a local temporary table. For more information about temporary tables, please see the section titled "Multiple SQL Statements and Local Temporary Tables".

Here is the list of the existing tables in the Soil Data Mart database for which public read access is granted. This table is sorted ascending on table physical name.

Table Physical Name	Table Label
area	Area
areatype	Area Type
chaashto	Horizon AASHTO
chconsistence	Horizon Consistence
chdesgnsuffix	Horizon Designation Suffix
chfrags	Horizon Fragments
chorizon	Horizon
chpores	Horizon Pores
chstruct	Horizon Structure
chstructgrp	Horizon Structure Group
chtext	Horizon Text
chtexture	Horizon Texture
chtexturegrp	Horizon Texture Group
chtexturemod	Horizon Texture Modifier
chunified	Horizon Unified
clippolygon	Clip Polygon
cocanopycover	Component Canopy Cover
cocropyld	Component Crop Yield
codiagfeatures	Component Diagnostic Features
coecoclass	Component Ecological Classification
coeplants	Component Existing Plants
coerosionacc	Component Erosion Accelerated
coforprod	Component Forest Productivity
coforprodo	Component Forest Productivity - Other
cogeomordesc	Component Geomorphic Description
cohydriccriteria	Component Hydric Criteria

cointerp	Component Interpretation
comonth	Component Month
component	Component
copm	Component Parent Material
copmgrp	Component Parent Material Group
copwindbreak	Component Potential Windbreak
corestrictions	Component Restrictions
cosoilmoist	Component Soil Moisture
cosoiltemp	Component Soil Temperature
cosurffrags	Component Surface Fragments
cosurfmorphgc	Component Three Dimensional Surface Morphometry
cosurfmorphhpp	Component Two Dimensional Surface Morphometry
cosurfmorphmr	Component Microrelief Surface Morphometry
cosurfmorphss	Component Slope Shape Surface Morphometry
cotaxfmmin	Component Taxonomic Family Mineralogy
cotaxmoistcl	Component Taxonomic Moisture Class
cotext	Component Text
cotreestomng	Component Trees To Manage
cotxfmother	Component Taxonomic Family Other Criteria
distinterpmd	Distribution Interpretation Metadata
distlegendmd	Distribution Legend Metadata
distmd	Distribution Metadata
featdesc	Feature Description
featline	Feature Line
featpoint	Feature Point
gsmmupolygon	General Soil Map Mapunit Polygon
laoverlap	Legend Area Overlap
legend	Legend
legendtext	Legend Text
mapunit	Mapunit
muaggatt	Mapunit Aggregated Attribute
muaoverlap	Mapunit Area Overlap
mucropyld	Mapunit Crop Yield
muline	Mapunit Line
mupoint	Mapunit Point
mupolygon	Mapunit Polygon
mutext	Mapunit Text

sacatalog	Survey Area Catalog
sainterp	Survey Area Interpretation
sapolygon	Survey Area Polygon
saspatialver	Survey Area Spatial Version
sastatusmap	Survey Area Status Map
satabularver	Survey Area Tabular Version
state	State

What Metadata Information is Available?

The metadata reports and diagrams described below are available at the following URL:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

Domains Report

A column's values may be restricted to a fixed set of ASCII values, which we refer to as a domain. All domains for all columns associated with a domain, for all tables, are included in this report. The domains in this report are sorted in ascending order on the report field titled "Domain Name". The value that is actually stored in a column associated with that domain is the value in the report column titled "Choice Value". The metadata report titled "Tables and Columns" indicates the name of a domain associated with a column, if any.

Table Column Descriptions Report

This report includes a description of each column in a table. The entries in this report are sorted in ascending order on the report field titled "Table Physical Name" and table column sequence, although table column sequence is not displayed in this report.

Tables and Columns Report

This report includes a row for each column in a table. For each table column, the following attributes are displayed:

> Column Sequence
> Column Physical Name
> Column Label
> Logical Data Type
> Physical Data
> Type Not Null? Size

Precision
Minimum Allowable Value
Maximum Allowable Value Units
of Measure
Associated Domain Name

For each table, a description of the purpose of that table is also included. The entries in this report are sorted in ascending order on the report field titled "Table Physical Name" and the report column titled "Seq".

Unique Constraints Report

This report includes all unique constraints for all tables. The rows in this report are sorted ascending on the report column titled "Table Physical Name", primary key constraint Boolean (not displayed in the report) and the report column titled "Constraint Name".

Table Relationships Diagram

This diagram shows all the relationships between all end-user queryable tables with the exception of tables "AREATYPE", "AREA" and "STATE". The diagram displays the physical column name or names on which any two tables are joined. The diagram also shows the cardinality of each relationship (one to one, one to zero or more, etc.).

What attributes can I depend on being populated?

In the Soil Data Mart database, very few columns are "Not Null". One of the reasons for this is that in our legacy source data, a lot of data was missing, and populating that data long after the initial soil survey was conducted is difficult and time consuming. Over time, we have added a lot of new attributes to the database, but those attributes don't tend to get populated until a major update of a survey area is performed.

Over time, the Soil Survey Division has provided our soil data stewards with guidelines as to which attributes should be populated by a certain point in time. At the current time, those guidelines indicate that, when appropriate, the following attributes should be populated. Please keep in mind that these guidelines are not enforced by any automated quality assurance process. Unless an attribute is denoted as "Not Null" in the Tables and Columns report, any use of that attribute must account for the possibility of that attribute's value being null.

For a description of any column listed below, please see the metadata report titled "Table Column Descriptions" at https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx.

Table Physical Name	Column Physical Name(s)	Qualifications
chfrags	fragvol_h	
chfrags	fragvol_l	
chfrags	fragvol_r	
chorizon	caco3_h	
chorizon	caco3_l	
chorizon	caco3_r	
chorizon	cec7_h	Required when representative pH >= 5.5.
chorizon	cec7_l	Required when representative pH >= 5.5.
chorizon	cec7_r	Required when representative pH >= 5.5.
chorizon	claytotal_r	
chorizon	dbfifteenbar	
chorizon	dbovendry_r	

Table Physical Name	Column Physical Name(s)	Qualifications
chorizon	dbthirdbar	
chorizon	desgnmaster	Required for an organic soil horizon or layer.
chorizon	ec_h	
chorizon	ec_l	
chorizon	ec_r	
chorizon	ecec_h	Required when representative pH < 5.5.
chorizon	ecec_l	Required when representative pH < 5.5.
chorizon	ecec_r	Required when representative pH < 5.5.
chorizon	gypsum_h	
chorizon	gypsum_l	
chorizon	gypsum_r	
chorizon	hzdepb_r	
chorizon	hzdept_r	
chorizon	hzname	
chorizon	kffact	Required for mineral soil horizons or layers. For organic soil horizons or layers, this value should be null.
chorizon	ksat_h	

chorizon	ksat_l	
chorizon	ksat_r	
chorizon	kwfact	Required for mineral soil horizons or layers. For organic soil horizons or layers, this value should be null.
chorizon	lep_r	
chorizon	om_h	
chorizon	om_l	
chorizon	om_r	
chorizon	ph01mcacl2_h	Required for organic soil horizons or layers.
chorizon	ph01mcacl2_l	Required for organic soil horizons or layers.
chorizon	ph01mcacl2_r	Required for organic soil horizons or layers.
chorizon	ph1to1h2o_h	Required for mineral soil horizons or layers.
chorizon	ph1to1h2o_l	Required for mineral soil horizons or layers.
chorizon	ph1to1h2o_r	Required for mineral soil horizons or layers.
chorizon	sandco_r	
chorizon	sandfine_r	
chorizon	sandmed_r	
chorizon	sandtotal_r	
chorizon	sandvc_r	
chorizon	sandvf_r	
chorizon	sar_h	
chorizon	sar_l	
chorizon	sar_r	

Table Physical Name	Column Physical Name(s)	Qualifications
chorizon	silttotal_r	
chorizon	sumbases_h	
chorizon	sumbases_l	
chorizon	sumbases_r	
chorizon	wfifteenbar_r	
chorizon	wtenthbar_r	Required for sandy soils.
chorizon	wthirdbar_r	Required for non-sandy soils.
chstruct	structgrade	A record is required for each representative structure kind.
chstruct	structsize	A record is required for each representative structure kind.

chstruct	structtype	A record is required for each representative structure kind.
chstructgrp	rvindicator	A record is required for the representative collection of structures.
chstructgrp	structgrpname	A record is required for the representative collection of structures.
chtexture	texcl or lieutex	One of the columns listed should be populated.
chtexturegrp	rvindicator	
chtexturegrp	texture	
cogeomordesc	geomfname	
cogeomordesc	geomftname	
cogeomordesc	rvindicator	
cohydriccriteria	hydriccriterion	A record is required for each qualifying criterion when the corresponding hydric rating is "Yes".
comonth	floddurcl	
comonth	flodfreqcl	
comonth	month	A record should exist for each month of the year.
comonth	ponddep_h	
comonth	ponddep_l	
comonth	ponddep_r	
comonth	ponddurcl	
comonth	pondfreqcl	
component	airtempa_h	
component	airtempa_l	
component	airtempa_r	
component	albedodry_r	
component	aspectccwise	
component	aspectcwise	
component	aspectrep	

Table Physical Name	Column Physical Name(s)	Qualifications
component	compkind	
component	compname	Required for soil and approved miscellaneous area map unit components.
component	comppct_r	

---

component	drainagecl	
component	ffd_h	
component	ffd_l	
component	ffd_r	
component	hydgrp	
component	hydricrating	
component	map_h	
component	map_l	
component	map_r	
component	slope_h	
component	slope_l	

component	slope_r	
component	taxgrtgroup	
component	taxorder	
component	taxsubgrp	
component	taxsuborder	
component	taxtempregime	
component	tfact	
component	weg	
component	wei	
copm	pmkind	A record is required for each representative kind of parent material.
copmgrp	pmgroupname	A record is required for the representative collection of parent materials.
copmgrp	rvindicator	A record is required for the representative collection of parent materials.
corestrictions	resdept_h	A record is required when either lithic bedrock or paralithic bedrock is present.
corestrictions	resdept_l	A record is required when either lithic bedrock or paralithic bedrock is present.
corestrictions	resdept_r	A record is required when either lithic bedrock or paralithic bedrock is present.
corestrictions	reskind	A record is required when either lithic bedrock or paralithic bedrock is present.
cosoilmoist	soimoistdept_h	A record is required for any layer whose corresponding soil moisture status is "wet".

Table Physical Name	Column Physical Name(s)	Qualifications
cosoilmoist	soimoistdept_l	A record is required for any layer whose corresponding soil moisture status is "wet".
cosoilmoist	soimoistdept_r	A record is required for any layer whose corresponding soil moisture status is "wet".
cosoilmoist	soimoiststat	A record is required for any layer whose corresponding soil moisture status is "wet".
cosurfmorphgc	geomposmntn or geomposhill or geompostrce or geomposflats	One of the columns listed should be populated.
cosurfmorphhpp	hillslopeprof	
cosurfmorphss	shapeacross	
cosurfmorphss	shapedown	
laoverlap	areasymbol	
laoverlap	areatypename	For SSURGO data, a record should exist for each geographic coincidence between the corresponding survey area and any instance of any of the following area types: MLRA (Major Land Resource Area), State or Territory, County and Parish. Records for "State or Territory" and "County or Parish" are guaranteed to exist. For STATSGO data, it appears that a record should exist for each geographic coincidence between the corresponding survey area and any instance of any of the following area types: MLRA (Major Land Resource), State or Territory.
legend	areasymbol	
legend	areatypename	
mapunit	muname	
mapunit	musym	

What fuzzy logic soil interpretations can I depend on being included?

In addition to the guidelines for which soil attributes should be populated in general, there are also certain fuzzy logic soil interpretations that should always be included. At the time that this was written, only the national military and Department of Homeland Security fuzzy logic soil interpretations are required to be included. Exactly which soil interpretations are included in those two groups may vary over time. Additional national and/or state specific fuzzy logic soil interpretations are typically included, but which of those soil interpretations are included may vary from survey area to survey area. At the current time, fuzzy logic soil interpretations are only available for map unit components.

The complete set of fuzzy logic soil interpretations that were included for a survey area is recorded in the table named "sainterp". The important things to keep in mind are:

1. Fuzzy logic soil interpretations are included on a survey area basis. The exact same set of fuzzy logic soil interpretations will be available for every map unit component in a given survey area.

2. When dealing with data that spans more than one survey area, the fuzzy logic soil interpretations that exist for some map unit components in that collection of data may not exist for all map unit components in that collection of data.

For a definition and discussion of fuzzy logic soil interpretations, please see the section titled "Interpretive Soil Attributes versus Fuzzy Logic Soil Interpretations".

Soil Attributes and Concepts

Soil Attribute Classification

Soil attributes can be grouped into a variety of classification schemes. This section discusses some of those potential classifications that aid in understanding and using the soil attributes that are represented in the Soil Data Mart database.

Empirical Soil Attributes versus Interpretive Soil Attributes

An empirical soil attribute is an attribute that is derived from observation or analysis, or an attribute whose value cannot be derived from other soil attributes. Slope, albedo, bulk density, pH and component name are examples of empirical soil attributes.

An interpretive soil attribute is a classification, or a rating for a specific use, that is derived, either logically or programmatically, from a function of one or more soil

attributes. Wind erodibility group, drainage class, T factor and texture class are examples of interpretive soil attributes that correspond to a classification. Suitability for a septic tank and suitability for camp areas are examples of interpretive soil attributes (fuzzy logic soil interpretations, actually) that correspond to a rating for a specific use.

To the degree possible, an interpretive soil attribute should be derived exclusively from empirical soil attributes, as opposed to being derived from a combination of empirical and interpretive soil attributes.

Interpretive Soil Attributes versus Fuzzy Logic Soil Interpretations

An interpretive soil attribute is similar to a fuzzy logic soil interpretation in that either may correspond to a classification, or a rating for a specific use. They differ by how they are populated or derived, and the number of columns that characterize the result.

An interpretive soil attribute is either manually populated or may be derived by an on demand calculation function. A typical interpretive soil attribute corresponds to a single column in a table where each record in that table corresponds to an instance of one of the soil entities represented in the Soil Data Mart database. These kind of interpretive soil attributes exist for map units, map unit components and soil horizons or layers. None of these kinds of interpretive soil attributes currently exist for a survey area.

A fuzzy logic soil interpretation is derived by a custom process that we refer to as the "fuzzy logic soil interpretation generation engine". This is a different on demand process from the one by which a calculated value is derived. An end-user does not have access to either of these on demand processes. A fuzzy logic soil interpretation result is characterized by multiple values. A fuzzy logic interpretive result is specified by one or more fuzzy logic values, i.e. a floating point value in the range zero to one. In addition, each of those fuzzy values is typically converted into a class. Lastly, in addition to the overall interpretive result, additional information about why that overall result was returned may be included. At the current time, fuzzy logic soil interpretations are only available for map unit components.

Temporal Soil Properties

Most soil attributes are not associated with a particular point in time or a particular time of the year, but some are. There are a number of soil attributes and concepts that are recorded by month. This set is composed of the following soil attributes:

Table and Column Physical Name	Description
comonth.flodfreqcl	Flooding Frequency
comonth.floddurcl	Flooding Duration
comonth.pondfreqcl	Ponding Frequency

comonth.ponddurcl	Ponding Duration
comonth.ponddep	Ponding Depth
comonth.dlyavgprecip	Mean Daily Average Precipitation
comonth.dlyavgpotet	Mean Daily Average Evapotranspiration
cosoilmoist.soimoistdept_l	Soil Moisture Depth to Top (Low)
cosoilmoist,soimoistdept_r	Soil Moisture Depth to Top (Representative)
cosoilmoist.soimoistdept_h	Soil Moisture Depth to Top (High)
cosoilmoist.soimoistdepb_l	Soil Moisture Depth to Bottom (Low)
cosoilmoist.soimoistdepb_r	Soil Moisture Depth to Bottom (Representative)
cosoilmoist.soimoistdepb_h	Soil Moisture Depth to Bottom (High)
Table and Column Physical Name	**Description**
cosoilmoist.soimoiststat	Soil Moisture Status
cosoiltemp.soitempmm	Mean Daily Average Temperature
cosoiltemp.soitempdept_l	Soil Temperature Depth to Top (Low)
cosoiltemp.soitempdept_r	Soil Temperature Depth to Top (Representative)
cosoiltemp.soitempdept_h	Soil Temperature Depth to Top (High)
cosoiltemp.soitempdepb_l	Soil Temperature Depth to Bottom (Low)
cosoiltemp.soitempdepb_r	Soil Temperature Depth to Bottom (Representative)
cosoiltemp.soitempdepb_h	Soil Temperature Depth to Bottom (High)

Soil Concepts

In the following section the term "concept" is used. How are "concepts" and soil attributes related? More than one soil attribute may be necessary to define a particular soil concept. In other words, for such concepts, an individual soil attribute or column may have no meaning in isolation.

Concepts that are represented by a Single Column in a Single Record

These are concepts that are adequately defined by a single soil attribute in isolation. This group includes most, if not all, soil attributes that reside in a table that has a one to one relationship with one of the four soil entities represented in the Soil Data Mart database. This set is composed of the following tables:

Table Physical Name	Soil Entity Represented by a Record in that Table
sacatalog	Survey Area
legend	Survey Area

mapunit	Map Unit
muaggatt	Map Unit
component	Map Unit Component
chorizon	Soil Horizon or Layer

Concepts that are represented by Multiple Columns in One or More Records in a Single Table

These are concepts whose definition typically requires more than one soil attribute in one or more records in a table whose records can repeat for one of the soil entities represented in the Soil Data Mart database. A single attribute in isolation may be inadequate to completely define a concept, and even multiple columns in a single record may not be adequate to completely define a concept. A couple of examples are probably in order.

A record in the table named "cocropyld" may include both an irrigated and a nonirrigated yield for a crop that is typically grown on the corresponding map unit component. The column named "cropname" records the name of the crop in question, the column named "yldunits" records the units in which that yield is expressed, and the column named "nonirryield" records the non-irrigated yield of that crop on the corresponding map unit component. Some of these columns, in isolation, are not particularly meaningful. In this case, a single record in this table does arguably completely define a concept.

A record in the table named "chfrags" corresponds to a particular kind, shape and size of fragment that occurs in the corresponding soil horizon or layer. Again, certain columns in this table in isolation are not particularly meaningful, and to fully characterize all fragments that occur in a particular soil horizon or layer may require multiple records.

We have indicated that, in general, this case pertains to attributes in records in tables that can repeat for their corresponding soil entity, but that is not always the case. With the exception of table "comonth", more than one record for the corresponding soil entity is ever required. In addition, some attributes in such a table may completely represent a concept. For example, cocropyld.cropname and chfrags.fragkind do completely identify a crop that tends to be grown on the corresponding map unit component, and a kind of fragment that occurs in the corresponding soil horizon or layer, respectively.

Fuzzy logic soil interpretations also fall into this group. For additional details about fuzzy logic soil interpretations, please see the section titled "Fuzzy Logic Soil Interpretations".

Concepts that are represented by Multiple Columns in Records in Multiple Tables

For certain complex soil concepts, multiple columns in records in multiple tables may be required in order to fully characterize that concept. This may be because some attributes that character that concept in part, repeat for other attributes that characterize that concept in part. This may also occur because for some concepts, in addition to the individual attributes that characterize that concept, we like to also provide a complex phrase that includes some or all of those individual soil attributes.

Soil texture class is an example of such a concept. A soil horizon or layer may be composed of multiple soil textures that occur in groups in that soil horizon or layer. More than one soil texture modifier may exist for a given soil texture class. Therefore, in this case, we are dealing with a three level hierarchy, soil texture modifier, soil texture class and soil texture class group. Each level of this hierarchy is stored in a different table in the database. For some purposes, it may be sufficient to display only the soil texture group (or groups) names. For some purposes, one may need access to each individual soil texture class in a corresponding soil texture class group.

Soil Concept	Physical Name of the Tables and Columns among which the Attributes that Characterize this Concept are Distributed	Table and Column Physical Name in which a Corresponding Complex Phrase is Recorded, if any
Map Unit Component Soil Taxonomic Classification	component.taxclname component.taxorder component.taxsuborder component.taxgrtgroup component.taxsubgrp component.taxpartsize component.taxpartsizemod component.taxceactcl component.taxreaction component.taxtempcl component.taxmoistscl component.taxtempregime component.soiltaxedition cotaxfmmin.* cotxfmother.* cotaxmoistcl.*	component.taxclname

Map Unit Component Geomorphology	component.geomdesc cogeomordesc.* cosurfmorphgc.* cosurfmorphhpp.* cosurfmorphmr.* cosurfmorphss.*	component.geomdesc (this phrase includes some, but not all, of the attributes involved)
Map Unit Component Hydric Rating	component.hydricrating cohydriccriteria.*	
Map Unit Component Soil Parent Material(s)	copmgrp.* copm.*	copmgrp.pmgroupname
Soil Horizon Nomenclature	chorizon.hzname chorizon.desgndisc chorizon.desgnmaster chorizon.desgnmasterprime chorizon.desgnvert chdesgnsuffix.*	chorizon.hzname
Soil Horizon or Layer Soil Texture	chtexturegrp.* chtexture.* chtexturemod	chtexturegrp.texture
Soil Horizon or Layer Soil Structure	chstructgrp.* chstruct.*	chstructgrp.structgrpname

Fuzzy Logic Soil Interpretations

Understanding Fuzzy Logic Soil Interpretations

When the information system that the Soil Survey Division uses to manage their transactional aggregated soil data was being design, we decided that we wanted a way to be able to continually add new, and modify existing, soil interpretations, without having to modify code in order to do so. We decided to add the infrastructure that would permit us to create new soil interpretation definitions, modify those soil interpretation definitions and then generate those soil interpretations, on demand.

We decided to do this by utilizing fuzzy logic, which on its surface, tends to strike people as pretty suspect. Explaining fuzzy logic is beyond the scope of this document, not to mention beyond the scope of my abilities. Suffice it to say that fuzzy logic can be utilized to determine the truth of a proposition, where the result, rather than being restricted to only True or False, is reported as a value in the range zero to one, inclusive. In effect, such a result is capable of dealing with shades of grey between True and False, or as Stephen Colbert has coined, the degree of "truthiness". The closer the value is to one, the closer the proposition is to being true.

For the typical fuzzy logic soil interpretation, the proposition is either that a soil is suitable for a particular use, or a soil is not suitable for a particular use. We refer to the former as a "suitability soil interpretation" and the latter as a "limitation soil interpretation". A fuzzy logic soil interpretation can be defined from either standpoint. For a suitability soil interpretation, the closer the result is to one, the more likely that soil is suitable for the corresponding use. For a limitation soil interpretation, the closer the result is to one, the more likely that soil is not suitable for the corresponding use. Whether a particular soil interpretation is a limitation or suitability soil interpretation is indicated by one of the attributes that is returned for that soil interpretation.

The typical soil interpretation is based on a group of sub-interpretations. While the result for a top level soil interpretation corresponds to the degree to which a soil is either suitable or unsuitable for a given use, the result for each sub-interpretation tends to correspond to whether or not a specific factor that makes that soil suitable or unsuitable for a particular use is present. For example, in determining the suitability of a soil for a dwelling with a basement, the sub-interpretation factors might include things like depth to bedrock and depth to water table. For a top level soil interpretation where the result is fully suitable, no sub-interpretation results are returned. For a top level soil interpretation where the result is anything less than fully suitable, the result for any sub-interpretation factor that contributed to the overall result being less than fully suitable is also returned, in order to document why the overall result was less than fully suitable.

The result for a soil interpretation is not a single value, and the reason why this is so takes some explaining. Let's start out by assuming that a single result is returned. That result would be a floating point value in the range zero to one, inclusive, that indicates the degree to which the corresponding soil is either suitable or unsuitable for the corresponding use. In addition to that floating point value, that floating point value is typically also converted into a class name that connotes the degree to which that soil is suitable or unsuitable for the corresponding use. For a suitability soil interpretation, the possible values of this class name might include "fully suitable", "marginally suitable" and "not suitable". For a limitation soil interpretation, the possible values of this class name might include "no limitations", "slight limitations" and "severe limitations". So at a minimum, a soil interpretation "result" will always include two values, one of which we refer to as the "rating value" and the other which we refer to as the "rating class name".

In reality, four different result pairs are always returned. The reason for this is related to the variability that can exist for a map unit component throughout a survey area. How the value of a soil property varies throughout a survey area for a given map unit component is expressed by recording most empirical soil property attributes as three different values - a value representing the low end of the range of that attribute for the corresponding map unit component throughout the corresponding survey area, a value representing the high end of the range of that attribute for the corresponding map unit component throughout the corresponding survey area and the typical or expected value of

that attribute for the corresponding map unit component throughout the corresponding survey area. This concept is more fully covered in the section titled "High, Low and Representative Values".

Because the value of a soil property can vary throughout a survey area, the result of a soil interpretation based on that soil property can also vary throughout a survey area. Therefore, for a soil interpretation, we also return multiple results in order to convey how the result for that soil interpretation can vary throughout the survey area. We return two pairs of results based on the two extremes of each soil property that serves as input to a soil interpretation. We also return a pair of results based on the range of the representative value of each soil property that serves as input to a soil interpretation.

It is not too difficult to understand why we might return three results pairs for a soil interpretation, one based on low values, one based on high values and one based on representative values, but it is not intuitively obvious why we return a pair of results based on representative values. After some debate, I decided to not explain the gory details behind this. I am hoping that it is sufficient to understand that four results pairs are always returned to characterize a soil interpretation result, and the reason that there are multiple results is to characterize how the result of that soil interpretation might vary throughout that survey area for specific instances of the corresponding map unit component.

We refer these four different result pairs as low/low, low/rv (representative value), high/rv and high/high. Each pair includes a rating value and typically a corresponding rating class name. Each rating value will always be a number in the range zero to one, inclusive, and the following will always be true:

low/low rating value <= low/rv rating value <= high/rv rating value <= high/high rating value

Which of these four values should you use? Since a result based on representative values is typically the best prediction of what one will encounter, we recommend using the rating values based on representative values. The low/low and high/high rating values are based on soil property values that are less likely to occur. In the end, it kind of comes down to whether you are a "glass is half full" person or a "glass is half empty" person.

For a suitability soil interpretation, the high/high rating value represents the most optimistic result, and the low/low rating value represents the most pessimistic result. For a suitability soil interpretation, the high/high value represents the most optimistic result because the closer to one, the more that soil is suitable for the corresponding use. For a suitability soil interpretation, we recommend using the low/rv rating value because it corresponds to the pessimistic or cautious side of the most probable result.

For a limitation soil interpretation, the low/low rating value represents the most optimistic result, and the high/high rating value represents the most pessimistic result. For a limitation soil interpretation, the high/high rating value represents the most pessimistic result because the closer to one, the more that soil is not suitable for the corresponding use. For a limitation soil interpretation, we recommend using the high/rv rating value because it corresponds to the pessimistic or cautious side of the most probable result.

The Storage and Organization of Fuzzy Logic Soil Interpretation Results

All fuzzy logic soil interpretation results are stored in the table named "cointerp". Records in this table repeat for their corresponding map unit component, because there are typically multiple different soil interpretations for a given map unit component. Records in this table may also repeat for a given soil interpretation for a given map unit component, because for any soil interpretation result that corresponds to less than fully suitable, not only will there be a record for that overall result, there will also be a record for each sub-interpretation factor that contributed to the overall result being less than fully suitable. In other words, this table is recursive in nature.

In a typical soil interpretation report available from either the Soil Data Mart (https://sdmdataaccess.sc.egov.usda.gov) or Web Soil Survey (https://websoilsurvey.sc.egov.usda.gov/), the report not only displays the rating class name corresponding to the overall result for a soil interpretation, but for any overall result corresponding to anything less than fully suitable, that report also displays the rating class name and rating value for each sub-interpretation factor that contributed to the overall result being less than fully suitable. Each sub-interpretation rating class names describes an individual reason why the overall rating was less than fully suitable, e.g. bedrock occurs within 60 inches, water table occurs within 60 inches, etc.

The rating values and rating class names displayed in a typical soil interpretation report correspond to the high/rv rating value and rating class name, regardless of whether the corresponding soil interpretation is a limitation soil interpretation or a suitability soil interpretation. This is contrary to the advice above that recommend using the low/rv rating value and rating class name for a suitability soil interpretation. This has been a long term oversight on our part that we hope to correct at some point.

This section obviously doesn't include everything one needs to know in order to be able to effectively query and work with fuzzy logic soil interpretation results. The remainder of that information is covered in detail in the sample query document titled "Returning Fuzzy Logic Soil Interpretations", available at the following URL:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

High, Low and Representative Values

The aggregated data that characterizes a survey area was partially derived from point samples taken over the course of that survey. A morphological description was recorded for a number of sample points, also known as pedons. Soil samples taken at numerous points were sent to a laboratory for a variety of physical, chemical and sometimes mineralogical analyses.

This data is eventually aggregated into a number of map unit components that characterize the soils in that survey area. Therefore the entire set of data for a survey area includes both point and aggregated data.

At this time the Soil Data Mart database includes only the aggregated data for a survey area. The point data that was collected in order to produce that aggregated data is currently stored in what we refer to as the National Soil Survey Center (NSSC) Soil Survey Laboratory Soil Characterization Database (https://ncsslabdatamart.sc.egov.usda.gov). At some point in time we hope to provide an integrated database that includes both point and aggregated data for a survey area, but for the time being, no point data is available in the Soil Data Mart database.

The value of many attributes of a map unit component can vary from location to location of that map unit component throughout the survey area. For this reason, for most empirical attributes, three values are recorded - a value representing the low end of the range of that attribute for the corresponding map unit component throughout the corresponding survey area, a value representing the high end of the range of that attribute for the corresponding map unit component throughout the corresponding survey area and the typical or expected value of that attribute for the corresponding map unit component throughout the corresponding survey area. We refer to the typical or expected value as the "representative value".

For such an attribute, the base part of the column name for all three values is the same. For the low, representative and high value, "_l", "_r" and "_h" are appended to the base part of the column name, respectively.

Representative Record(s)

For some collections of attributes that repeat for their corresponding soil entity, one or more of those collections, or records, may be indicated as being "representative". This means that for the soil entity in question, the values of the attributes in those repeating group records that are marked as being representative, are typical for an instance of that soil entity. Such a record will include a column named "rvindicator". Column "rvindicator" contains a Boolean value that indicates whether or not the collection of attributes values in that record is considered to be representative for the corresponding

soil entity. Although more than one collection can be marked a representative, in most cases, only one collection will be marked as representative. Unfortunately, no automated quality assurance process ensures that, for such a collection, at least one collection is marked as representative.

A record in any of the following tables will always include a column named "rvindicator" whose value indicates whether or not that record is considered to be representative for the corresponding soil entity.

Table Physical Name	Corresponding Soil Entity/Table Physical Name
chaashto	Soil Horizon or Layer/chorizon
chconsistence	Soil Horizon or Layer/chorizon
chpores	Soil Horizon or Layer/chorizon
chstructgrp	Soil Horizon or Layer/chorizon
chtexturegrp	Soil Horizon or Layer/chorizon
chunified	Soil Horizon or Layer/chorizon
coerosionacc	Map Unit Component/component
cogeomordesc	Map Unit Component/component
copmgrp	Map Unit Component/component

Case Sensitivity

In the end-user queryable tables in the Soil Data Mart database, with two exceptions, all columns whose value may include one or more letters are case insensitive. In the entire Soil Data Mart database, only columns mapunit.musym and muaggatt.musym are case sensitive.

So when a query references a literal value for the vast majority of columns, case doesn't matter, but whenever a query references a literal map unit symbol, the case of the letters in that map unit symbol, if any, must be correct.

Discriminating between SSURGO and STATSGO

SSURGO is an acronym that corresponds to our detailed soil map data. The map scale for a typical SSURGO survey area is 1:24000. There are over 3000 SSURGO survey areas in the Soil Data Mart. For the definition of a survey area, see the section titled "The Soil Related Entities in the Soil Data Mart Database". The SSURGO data spans the 50 U.S. states, the District of Columbia, Puerto Rico and the Virgin Islands, and Guam and other Pacific islands that either use to or still comprise what is referred to as the United States Trust Territories.

For any SSURGO survey area, legend.areatypename = 'Non-MLRA Soil Survey Area'. For any SSURGO survey area, legend.areasymbol will always contain five characters where the first two characters denote the state or territory (FIPS alphabetic postal code) with administrative responsibility for the corresponding survey area, and the last three characters are a three digit zero filled integer value used to distinguish survey areas for the same state or territory.

STATSGO is an acronym that corresponds to the United States general soil map data. For STATSGO there is only one survey area whose map scale is approximately 1:250,000. STATSGO spans the 50 U.S. states, the District of Columbia, Puerto Rico and the Virgin Islands. For the single STATSGO survey area, legend.areatypename = 'Country' and legend.areasymbol = 'US'.

One idiosyncrasy of the Soil Data Mart database is that while the spatial data for SSURGO and STATSGO is stored in a different set of tables, the non-spatial or tabular data for both SSURGO and STATSGO is stored in exactly the same set of tables. This means than unless you want tabular data for both SSURGO and STATSGO to be returned for a given query, which for most purposes would be unlikely, any query for tabular data must explicitly distinguish between SSURGO and STATSGO.

The can be done by including table "legend" in your query and either explicitly including or excluding certain values for legend.areatypename and legend.areasymbol.

The sample query document titled "Discriminating between SSURGO and STATSGO" discusses this topic in detail and provides a number of sample queries. This document is available at the following URL:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

Data Version and Citing the Soil Data Mart

The data in the Soil Data Mart is the most current data available at the time it is accessed. Internally, our data is managed and updated on a survey area basis. For every survey area in the Soil Data Mart database there is a corresponding version number and date/time stamp. Although the Soil Data Mart contains only the data for the most recent version of a survey area, the data for every version of a survey area that was ever released, at least since we started formally versioning our data, is available in what we refer to as the Soil Data Warehouse. We started formally versioning our data sometime in December 2003.

Data for a survey area is updated over time, and when updated data for a survey area is released, a new version number and date/time stamp are assigned. Survey area version numbers are assigned sequentially. It is not uncommon for the data for a survey area to be updated more than once a year. Soil Data Mart clients can subscribe to a survey area

in order to be automatically notified whenever a new version of data for that survey area is released.

For a conservation assistance contract, or for a wetland or highly erodible land determination, there may be a requirement to record the version of data on which that contract or determination was based. One might want to associate the results of an analysis with the version numbers of any survey areas whose data participated in that analysis. This can be accomplished by, for each survey area, recording either a survey area's version number or its associated date/time stamp. For a given date and time, survey area version can be derived, and the time interval for which a given survey area version was current can also be derived.

To cite a version of a survey area, we suggest something like:

> Soil Survey Staff, Natural Resources Conservation Service, United States Department of Agriculture. Soil survey of Jefferson County Area and Part of Silver Bow County, Montana [Online WWW]. Available URL: "https://websoilsurvey.sc.egov.usda.gov", State="MT" [Accessed 29 August 2016].

Where "Jefferson County Area and Part of Silver Bow County, Montana" is the name in the column titled "Survey Area Name" at https://websoilsurvey.sc.egov.usda.gov, and "MT" is the state postal code of the state for which that survey area is listed. By specifying the date accessed the corresponding survey area version number can be determined. It would also be acceptable to substitute the survey area version number or data/time stamp in lieu of the date when that data was accessed.

A survey area that spans states is listed for both states, and the corresponding survey area name typically reflects both states. For such a case, it doesn't matter which state postal code is reflected in the URL.

Executing Queries against the Soil Data Mart Database

Introduction

One of our challenges was to come up with a way to let well-intentioned end-users execute queries against the Soil Data Mart database, but prevent well-intentioned end users from inadvertently, and malicious miscreants from intentionally, altering the database or the server on which the database resides. We have attempted to do so by ultimately using a SQL Server login with restricted permissions, setting the appropriate

SQL Server and database permissions, and by pre-scanning any query for any metacharacters and keywords that we have deemed should be restricted. Whether or not we will be successful in preventing any damage to the database remains to be seen.

There are three ways to execute a query against the Soil Data Mart database:

1. Submit a query via our web service.

2. Submit a query via our web application query interface that executes a query in real-time and returns the results to your browser. This option is referred to as "IMMEDIATE QUERY". To execute a query immediately, click one of the buttons whose corresponding label begins with the word "Immediate", before submitting your query.

3. Submit a query via our web application query interface that queues your request, executes it when appropriate, and sends you an e-mail notification when your query has either completed, timed out or has been rejected for any reason. This option is referred to as "QUEUED QUERY". To submit a query to be queued and run later, click one of the buttons whose corresponding label begins with the word "Queued", before submitting your query.

In addition to protecting the database and database server from inadvertent or malicious alteration, we also needed to come up with some sort of scheme to prevent our database server's system resources from being exhausted. We have attempted to do this by limiting the time a query can run, and by processing queued query requests serially rather than in parallel.

The amount of time that a query is allowed to run is referred to as a timeout value. The timeout value for a query whose results are returned in real-time is shorter than the timeout value for a query submitted to the queue. These timeout parameter values are not permanently fixed because system resources can change over time and because we have little or no idea what the request load is actually going to be over time. Unfortunately, you have no way of knowing whether or not a query you submit is going to timeout. A given query could timeout one time but not another time, due to competition for system resources at any given moment.

A query submitted via "IMMEDIATE QUERY" can complete within the current timeout limit but still return too many records to be returned to your browser in real-time. Therefore in addition to a shorter timeout value for immediate queries, there is also a limit on the total number of records that can be returned to your browser. This limit is a parameter whose value can change over time, depending on system resources. If the number of rows returned by an immediate query exceeds this limit, a message will be displayed informing you that this has occurred and the message will indicate that the

query in question should be submitted as a queued query as opposed to an immediate query.

For a query submitted via a web service, the results are always returned in XML.

For a query submitted via our real-time web application interface, results can be returned in either XML or HTML.

For a queued query submitted via our non-real-time web application interface, results can be returned in an ASCII file either as XML or as field delimited text. For data returned in an ASCII delimited text file, the user can select from a fixed set of field delimiters and optional text delimiters, and the user can also indicate whether or not the first line in that file should contain column names. Any column alias defined in a Select statement will be returned as the corresponding column name, when column names are requested.

Unconstrained Queries

Our intention is to support requests capable of returning every soil horizon or layer for every map unit component for every map unit for every survey area in the database. We want to provide end-users with a way to request soil tabular data for a large geographic extent, where all of the data returned for that request is bundled together, rather than by survey area, which is a current limitation of soil data exports from either the Soil Data Mart web application or the Geospatial Data Gateway web application.

On the other hand, all queries are constrained by a timeout value. It is unlikely that an unconstrained query submitted via a web service or the real-time query interface will complete within the limited timeout value allowed for a real-time request. In general, requests for large volumes of data should be submitted as a queued query. Even then, the number of attributes that can be returned by such a request is limited. When submitting an unconstrained query, you will need to be discerning about which attributes you select, because ultimately, the timeout value allowed for a queued query request is going to force you to be discerning.

Creating and Editing a Query

The queries in the sample query documents were originally created in a Microsoft Word document. These queries originally failed because the Microsoft SQL Server query processor doesn't recognize the directional quotes that Word inserted by default. By default, MS Word also converts two sequential dashes into either an en dash or an em dash, which means that that what you intended to be treated as a comment won't be recognized as a comment by the Microsoft SQL query processor. These automatic behaviors can be disabled by changing the corresponding options in the Replace section

of the AutoFormat tab via Tools:AutoCorrect Options. There may be other issues that I haven't encountered yet.

In general, it would be better if you used something other than Microsoft Word to create and edit a query. Notepad doesn't have the issues discussed above. I have no idea what might be available for a Macintosh. I prefer to use a third party ASCII editor, UltraEdit32 in my case, but there is typically a charge for one of these products.

Query Errors

One limitation of our initial implementation is that when your query contains an error, we are unable to provide any diagnostic that indicates the location of an error. All we can tell you is that your query contains an error of some sort. By and large, it is entirely up to you to diagnose and correct any errors. As a last resort, we can provide some support, but we ask that you make an initial attempt to find the source of an error. At some point in the future, we may be able to provide some sort of diagnostic output.

## Multiple SQL Statements and Local Temporary Tables

Up to now we have been speaking in terms of submitting "a" query, which might be natural to interpret as a single Select statement. You can actually submit more than one SQL statement in a request, keeping in mind that all statements must complete within a timeout value in order for anything to be returned. Not all of the statements in a request have to be Select statements, and nothing prevents you from submitting more than one Select statement in a given request. All that notwithstanding, there is probably no legitimate reason that someone would ever want to submit a request where the last SQL statement in that request was not a Select statement.

What other types of SQL statements might one want to use? For example, we allow you to create local temporary tables and insert records into those tables. Our original reason for permitting the use of local temporary tables was in order to provide a way to improve the performance of a query whose results are constrained to an area of interest, where that constraint references a large number of explicit geographic regions. For further details on this subject, please see the section titled "Constraining a Query to an Area of Interest".

Note that while local temporary tables are permitted, global temporary tables are not permitted. The reason for this is to prevent a huge temporary table from being created over multiple overlapping requests.

Data Bundling, Data Format, and Multiple Select Statements in a Request

The examples that follow are based on the following data model, where each table contains two records. The arrows in the following diagram point to the corresponding child table's parent table.
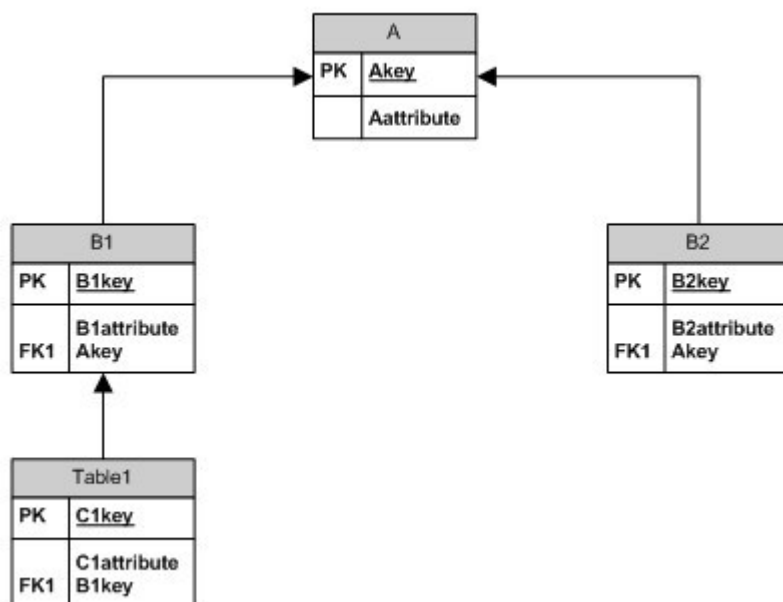
The results that follow are all based on the following SQL script.

```
SELECT Aattribute, B1attribute, C1attribute, A.Akey, B1.B1key, C1key
FROM A
  INNER JOIN B1 ON A.Akey = B1.Akey
   INNER JOIN C1 ON B1.B1key = C1.B1key


SELECT Aattribute, B2attribute, A.Akey, B2key
FROM A
  INNER JOIN B2 ON A.Akey = B2.Akey
```

HTML Format

If you have requested your results in HTML, which can only be done using the IMMEDIATE QUERY option of our web application query interface, a separate HTML table is returned for each Select statement included in the request.

Aattribute	B1attribute	C1attribute	Akey	B1key	C1key
Aattribute1	B1attribute1	C1attribute1	Akey1	B1key1	C1key1
Aattribute2	B1attribute2	C1attribute2	Akey2	B1key2	C1key2

Aattribute	B2attribute	Akey	B2key
Aattribute1	B2attribute1	Akey1	B2key1
Aattribute2	B2attribute2	Akey2	B2key2

XML Format

If you have requested your results in XML, regardless of how that request was submitted, web service or IMMEDIATE QUERY or QUEUED QUERY, the combined result set for all Select statements included in a request, will be delimited by the tag "<NewDataSet>". Results for the first Select statement correspond to the tag "<Table>". Results for each successive Select statement correspond to the tag "<TableN>", when N is the Select statement's sequence, minus one. In other words, the tag for results for the second Select statement is "<Table1>".

```
- <NewDataSet>
  - <Table>
      <Aattribute>Aattribute1</Aattribute>
      <B1attribute>B1attribute1</B1attribute>
      <C1attribute>C1attribute1</C1attribute>
      <Akey>Akey1</Akey>
      <B1key>B1key1</B1key>
      <C1key>C1key1</C1key>
    </Table>
  - <Table>
      <Aattribute>Aattribute2</Aattribute>
      <B1attribute>B1attribute2</B1attribute>
      <C1attribute>C1attribute2</C1attribute>
      <Akey>Akey2</Akey>
      <B1key>B1key2</B1key>
      <C1key>C1key2</C1key>
    </Table>
  - <Table1>
      <Aattribute>Aattribute1</Aattribute>
      <B2attribute>B2attribute1</B2attribute>
      <Akey>Akey1</Akey>
      <B2key>B2key1</B2key>
    </Table1>
  - <Table1>
      <Aattribute>Aattribute2</Aattribute>
      <B2attribute>B2attribute2</B2attribute>
      <Akey>Akey2</Akey>
      <B2key>B2key2</B2key>
    </Table1>
</NewDataSet>
```

ASCII Delimited Text File Format

If you have requested your results in ASCII delimited text file format, which can only be done using the QUEUED QUERY option of our web application query interface, the entire result set for a given request is returned in a single zip file. This zip file will include an individual file for each Select statement included in the query script.

Results for the first Select statement are returned in a file named "Table". Results for each successive Select statement are returned in a file named "TableN", when N is the Select statement's sequence, minus one. In other words, the name of the file containing the results for the second Select statement is "Table1".

For the sample results shown below, the "First row contains column names" option was checked, the field delimiter was set to the vertical bar character and the text delimiter was set to the double quote character. For each result set shown below, the first line contains column names. This is not real obvious because the actual attribute values are variants of column names.

Contents of file "Table.txt":

"Aattribute"|"B1attribute"|"C1attribute"|"Akey"|"B1key"|"C1key"
"Aattribute1"|"B1attribute1"|"C1attribute1"|"Akey1"|"B1key1"|"C1key1"
"Aattribute2"|"B1attribute2"|"C1attribute2"|"Akey2"|"B1key2"|"C1key2"

Contents of file "Table1.txt":

"Aattribute"|"B2attribute"|"Akey"|"B2key"
"Aattribute1"|"B2attribute1"|"Akey1"|"B2key1"
"Aattribute2"|"B2attribute2"|"Akey2"|"B2key2"

ADO Shape Command

A query submitted to execute against the Soil Data Mart database may include either of the following:

1. A SQL script that includes one or more Select statements.

2. A script that includes one and only one Shape command.

A query submitted to be executed against the Soil Data Mart database cannot include both a Select statement and a Shape command.

A traditional Select statement returns a single relation (a single set of rows and columns). A Shape command is capable of returning a hierarchical dataset, i.e. a dataset that includes multiple relations. In what circumstances might it be preferable to use a Shape command rather than a Select statement?

A traditional Select statement that returns data from multiple database tables returns a single table where certain information in that table is typically duplicated. Information about the original organizational structure of that data is lost. For some uses of that resulting data, the duplication is relatively easily dealt with, but if one wanted to extract data from one database and then insert that data into another database with the same organizational structure as the original database, that task is much easier if the information about the original organizational structure is preserved.

Another issue is related to the problem of Cartesian products. When a query returns data for a single branch in a table hierarchy, the resulting duplication is relatively straight forward to deal with. If a query includes data from tables in a hierarchy that represent more than a single branch, the duplication in the resulting Cartesian product is much more difficult to work with. Returning the results of such a query as a hierarchical dataset may reduce the complexity of working with that data, but on the other hand, making effective use of a hierarchical dataset also requires a higher level of expertise.

The typical end-user probably has little use for hierarchical datasets. This option is primarily intended for advanced users who are likely to be writing their own programs to process the data returned from their requests. For most users it is probably going to be enough of a challenge to learn the syntax for a traditional Select statement, much less the syntax for a Shape command.

Below is a simple sample Shape command that returns map unit symbol, map unit name, component name and corresponding representative percent composition for all components for all map units for the Hall County Nebraska SSURGO survey area.

```
SHAPE {
SELECT musym, muname, mukey
FROM legend l
  INNER JOIN mapunit mu ON l.lkey = mu.lkey
WHERE areasymbol = 'NE079'
ORDER BY museq}
APPEND ({
SELECT mu.mukey, comppct_r, compname
FROM legend l
  INNER JOIN mapunit mu ON l.lkey = mu.lkey
    LEFT OUTER JOIN component c ON mu.mukey = c.mukey
WHERE areasymbol = 'NE079'
ORDER BY comppct_r DESC, compname}
AS component RELATE mukey TO mukey)
```

For additional information on constructing and using Shape commands, please see a current text on Active X Data Objects (ADO), or search the web using the key phrase "ADO shape command". At the time this was written, an introduction to Shape command syntax was available as an article in the Microsoft Knowledge Base at the following URL:

http://support.microsoft.com/kb/189657/en-us

A couple sample Shape commands are also included in the sample query document titled "Using an ADO Shape Command", available at the following URL:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

Data Bundling and Data Format for the Results of a Shape Command

The examples that follow are based on the following data model, where each table contains two records. The arrows in the following diagram point to the corresponding child table's parent table.
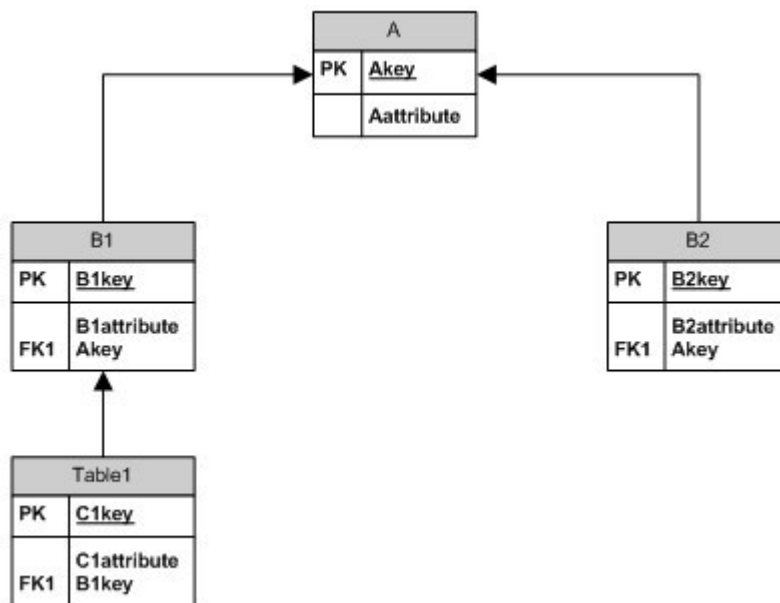
The results that follow are all based on the following Shape command.

```
SHAPE {
SELECT * FROM A}  APPEND
((SHAPE {
    SELECT * FROM B1}
    APPEND ({SELECT * FROM C1} AS C1
        RELATE b1key TO b1key)) AS B1
    RELATE akey TO akey),
   ({SELECT * FROM B2} AS B2
    RELATE akey TO akey)
```

HTML Format

If you have requested your results in HTML, which can only be done using the
IMMEDIATE QUERY option of our web application query interface, a separate HTML
table is returned for each Select statement included in the Shape command. Note that the
Shape command processor has added extra columns based on the name following the As
keyword associated with each Select statement but the first.

Aattribute	Akey	B1	B2
Aattribute1	Akey1	0	0
Aattribute2	Akey2	1	1

B1attribute	B1key	Akey	C1	TableB1
B1attribute1	B1key1	Akey1	0	0
B1attribute2	B1key2	Akey2	1	1

C1attribute	C1key	B1key	TableB1C1
C1attribute1	C1key1	B1key1	0
C1attribute2	C1key2	B1key2	1

B2attribute	B2key	Akey	TableB2
B2attribute1	B2key1	Akey1	0
B2attribute2	B2key2	Akey2	1

XML Format

If you have requested your results in XML, regardless of how that request was submitted, web service or IMMEDIATE QUERY or QUEUED QUERY, the combined result set for all Select statements included in the Shape command, will be delimited by the tag "<NewDataSet>". Results for the first Select statement correspond to the tag "<Table>". Results for each successive Select statement correspond to a tag whose name is based on the name following the As keyword associated with each successive Select statement. The tag names for Select statements below the second level of the select hierarchy reflect a concatenation of all As keyword names up to the second level of the hierarchy. For example, the third Select statement in the Shape command shown above corresponds to the third level of the select hierarchy. Therefore the tag name, <TableB1C1>, includes both "B1" and "C1".

```
- <NewDataSet>
  - <Table>
      <Aattribute>Aattribute1</Aattribute>
      <Akey>Akey1</Akey>
      <B1>0</B1>
      <B2>0</B2>
   </Table>
  - <Table>
      <Aattribute>Aattribute2</Aattribute>
      <Akey>Akey2</Akey>
      <B1>1</B1>
      <B2>1</B2>
   </Table>
  - <TableB1>
      <B1attribute>B1attribute1</B1attribute>
      <B1key>B1key1</B1key>
      <Akey>Akey1</Akey>
      <C1>0</C1>
      <TableB1>0</TableB1>
   </TableB1>
  - <TableB1>
      <B1attribute>B1attribute2</B1attribute>
      <B1key>B1key2</B1key>
      <Akey>Akey2</Akey>
      <C1>1</C1>
      <TableB1>1</TableB1>
   </TableB1>
  - <TableB1C1>
      <C1attribute>C1attribute1</C1attribute>
      <C1key>C1key1</C1key>
      <B1key>B1key1</B1key>
      <TableB1C1>0</TableB1C1>
   </TableB1C1>
  - <TableB1C1>
      <C1attribute>C1attribute2</C1attribute>
      <C1key>C1key2</C1key>
      <B1key>B1key2</B1key>
      <TableB1C1>1</TableB1C1>
   </TableB1C1>
  - <TableB2>
      <B2attribute>B2attribute1</B2attribute>
      <B2key>B2key1</B2key>
      <Akey>Akey1</Akey>
      <TableB2>0</TableB2>
   </TableB2>
  - <TableB2>
      <B2attribute>B2attribute2</B2attribute>
      <B2key>B2key2</B2key>
      <Akey>Akey2</Akey>
      <TableB2>1</TableB2>
   </TableB2>
</NewDataSet>
```

ASCII Delimited Text File Format

If you have requested your results in ASCII delimited text file format, which can only be done using the QUEUED QUERY option of our web application query interface, the

entire result set for a given request is returned in a single zip file. This zip file will include an individual file for each Select statement included in the Shape command.

Results for the first Select statement are returned in a file named "Table.txt". The name of the result file for each successive Select statement is based on the name following the As keyword associated with each successive Select statement. The table names for Select statements below the second level of the select hierarchy reflect a concatenation of all As keyword names up to the second level of the hierarchy. For example, the third Select statement in the Shape command shown above corresponds to the third level of the select hierarchy. Therefore the table name, "TableB1C1.txt", includes both "B1" and "C1".

For the sample results shown below, the "First row contains column names" option was checked, the field delimiter was set to the vertical bar character and the text delimiter was set to the double quote character. For each result set shown below, the first line contains column names. This is not real obvious because the actual attribute values are variants of column names.

Contents of file "Table.txt":

"Aattribute"|"Akey"|"B1"|"B2" "Aattribute1"|"Akey1"|0|0
"Aattribute2"|"Akey2"|1|1

Contents of file "TableB1.txt":

"B1attribute"|"B1key"|"Akey"|"C1"|"TableB1"
"B1attribute1"|"B1key1"|"Akey1"|0|0
"B1attribute2"|"B1key2"|"Akey2"|1|1

Contents of file "TableB1C1.txt":

"C1attribute"|"C1key"|"B1key"|"TableB1C1"
"C1attribute1"|"C1key1"|"B1key1"|0
"C1attribute2"|"C1key2"|"B1key2"|1

Contents of file "TableB2.txt":

"B2attribute"|"B2key"|"Akey"|"TableB2"
"B2attribute1"|"B2key1"|"Akey1"|0
"B2attribute2"|"B2key2"|"Akey2"|1

Reserved Meta-Characters and Keywords

The following meta-characters and keywords cannot be used outside the context of a quoted literal in the text of a query. If any of these reserved meta-characters or keywords

is detected outside the context of a quoted literal, the corresponding query will not be executed, and your response should indicate why your query was rejected.

##	@@	begin	browse	bulk
current_user	db_name	declare	exec	execute
file_name	goto	has_dbaccess	holdlock	host_id
host_name	is_member	is_srvrolemember	master	msdb
opendatasource	openquery	openrowset	option	paglock
parsename	repeatableread	rowlock	serializable	serverproperty
session_user	sessionproperty	set	suser_sid	suser_sname
system_user	tablock	tablockx	updatetext	updlock
user	user_id	user_name	waitfor	while
writetext	xlock			

Constraining a Query to an Area of Interest

An end-user is not required to constrain a query's results to some area of interest. In fact an end-user is not required to constrain a query's results by any means, i.e. for any Select statement, no Where clause is required. Keep in mind though that the total volume of data that can be returned is still ultimately constrained by that query's associated timeout value.

All of this notwithstanding, in many cases an end-user will want to constrain a query's results to some area of interest. This can only be done by referencing certain geographic regions that are guaranteed to always be available in the tabular data, where those geographic regions share some spatial coincidence with your area or interest.

For SSURGO survey area data, the following entities that correspond to a set of polygons that make up a logical geographic region are guaranteed to always be available in the tabular data:

Map unit

Survey area

County or parish that shares a spatial coincidence with the corresponding survey area, in whole or part

State or territory that shares a spatial coincidence with the corresponding survey area, in whole or part

For STATSGO survey area data, the only entity that corresponds to a set of polygons that make up a logical geographic region that is guaranteed to always be available in the tabular data is a map unit.

To constrain a query's results to an area of interest, one must first determine a set of logical geographic regions that share a coincidence with the area of interest, in whole or part, and then use references to those logical geographic regions as a constraint. The smaller the logical geographic region that is used for this constraint, the more accurately the results will be constrained to the desired area of interest. The only way to constrain results to an area of interest with complete accuracy is to reference only map units. Using any other logical geographic region is a compromise that may result in the inclusion of data that lies outside the area of interest.

The sample query document titled "Constraining a Query to an Area of Interest" discusses this topic in detail and provides a number of sample queries. This document is available at the following URL:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

Where can I find some sample queries to help me get started?

A number of sample queries are provided in a set of PDF documents that are available on the Query Help page of the SDM Data Access website, which is also where the document you are currently reading resides. That URL is:

https://sdmdataaccess.sc.egov.usda.gov/queryhelp.aspx

The sample query document titled "Fundamental Query" illustrates one variant of the most common query that will be executed against the Soil Data Mart database. That document lays the foundation for understanding all other sample query documents. The initial list of sample query documents includes:

Fundamental Query

Discriminating between SSURGO and STATSGO

Constraining a Query to an Area of Interest

Returning Soil Texture Related Attributes

Returning Fuzzy Logic Soil Interpretations

Using an ADO Shape Command

Prior to creating your own queries to execute against the Soil Data Mart Database, at an absolute minimum, you are strongly advised to read the sample query documents titled "Fundamental Query", "Discriminating between SSURGO and STATSGO" and "Constraining a Query to an Area of Interest".